IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
(Case No. RA043D2C15)

In the Application of:	)
FARMWALD ET AL.	)
Serial No: Continuation of 09/545,648	)
Filed: Herewith	)
Title: MEMORY DEVICE HAVING A PROGRAMMABLE REGISTER (As Amended)	)

Assistant Commissioner for Patents
Washington, DC 20231

PRELIMINARY AMENDMENT

Dear Sir:

Prior to the examination of the above-referenced application, kindly amend the application as follows:

IN THE ABSTRACT:

Please delete the Abstract of the Disclosure and substitute the attached Abstract of the Disclosure.

IN THE TITLE:

Please delete the title and substitute --MEMORY DEVICE HAVING A PROGRAMMABLE REGISTER--.

IN THE SPECIFICATION:

(A marked-up version of the following amendments to the Specification is enclosed herewith)

On page 1, line 8, **insert**:

This application is a continuation of Application No. 09/545,648, filed on April 10, 2000 (still pending), which is a continuation of Application No. 09/161,090, filed on September 25, 1998 (now U.S. Patent 6,049,846), which is a division of Application No. 08/798,520, filed on February 10, 1997 (now U.S. Patent 5,841,580), which is a division of Application No. 08/448,657, filed May 24, 1995 (now U.S. Patent 5,638,334), which is a division of Application No. 08/222,646, filed on March 31, 1994 (now U.S. Patent 5,513,327), which is a continuation of Application No. 07/954,945, filed on September 30, 1992 (now U.S. Patent 5,319,755), which is a continuation of Application No. 07/510,898, filed on April 18, 1990 (now abandoned).

On page 3, **substitute** the paragraph starting on line 6 with the following paragraph:

Prior art memory systems have attempted to solve the problem of high speed access to memory with limited success. U.S. Pat. No. 3,821,715 (Hoff et.al.), was issued to Intel Corporation for the earliest 4-bit microprocessor. That patent describes a bus connecting a single central processing unit (CPU) with multiple RAMs and ROMs. That bus multiplexes addresses and data over a 4-bit wide bus and uses point-to-point control signals to select particular RAMs or ROMs. The access time is fixed and only a single processing element in permitted. There is no block-mode type of operation, and most important, not all of the interface signals between the devices are bused (the ROM and RAM control lines and the RAM select lines are point-to-point).

On page 6, **substitute** the paragraph starting on line 1, with the following paragraph:

In U.S. Pat. No. 4,646,270 (Voss), a video RAM is described which implements a parallel-load, serial-out shift register on the output of a DRAM. This generally allows greatly improved bandwidth (and has been extended to 2, 4 and greater width shift-out paths.) The rest of the interfaces to the DRAM (RAS, CAS, multiplexed address, etc.) remain the same as for conventional DRAMS.

On page 10, **substitute** the paragraphs starting on lines 18 and 21 , with the following two paragraphs, respectively:

FIGS. 7a and 7b show the timing whereby signals from two devices can overlap temporarily and drive the bus at the same time.

FIGS. 8a and 8b show the connection and timing between bus clocks and devices on the bus.

On page 11,starting at line 14, **insert** the following paragraph:

Figure 16 is a block diagram representation of a set of internal registers within each device illustrated in Figure 2.

On page 14, **substitute** the paragraph starting on line 3, with the following paragraph:

With reference to FIG. 16, each semiconductor device contains a set of internal registers 170, preferably including a device identification (device register 171, a device-type descriptor register 174, control registers 175 and other registers containing other information relevant to that type of device. In a preferred implementation, semiconductor devices connected to the bus contain registers 172 which specify the memory addresses contained within that device and access-time registers 173 which store a set of one or more delay times at which the device can or should be available to send or receive data.

On page 14, **substitute** the paragraph beginning on line 13 and ending on page 15, line 3, with the following paragraph:

Most of these registers can be modified and preferably are set as part of an initialization sequence that occurs when the system is powered up or reset. During the initialization sequence each device on the bus is assigned a unique device ID number, which is stored is the device ID register 171. A bus master can then use these device ID numbers to access and set appropriate registers in other devices, including access-time to registers 173, control registers 175, and memory registers 172, to configure the system. Each slave may have one or several access-time registers 173 (four in a preferred embodiment). In a preferred embodiment, one access-time register in each slave is

permanently or semi-permanently programmed with a fixed value to facilitate certain control functions. A preferred implementation of an initialization sequence is described below in more detail.

On page 21, **substitute** the paragraph starting on line 8, with the following paragraph:

The data block transfer occurs later at a time specified in the request packet control information, preferably beginning on an even cycle. A device begins a data block transfer almost immediately with a device-internal phase as the device initiates certain functions, such as setting up memory addressing, before the bus access phase begins. The time after which a data block is driven onto the bus lines is selected from values stored in slave access-time registers 173. The timing of data for reads and writes in preferably the same; the only difference is which device drives the bus. For reads, the slave drives the bus and the master latches the values from the bus. For writes the master drives the bus and the selected slave latches the values from the bus.

On page 34, **substitute** the paragraph starting on line 4, with the following paragraph:

Slave devices do not need to detect a collision directly, but they must wait to do anything irrecoverable until the last byte (byte 5) is read to ensure that the packet is valid. A request packet with Master[0:3] equal to 0 (a retry signal) is ignored and does not cause a collision. The subsequent bytes of such a packet are ignored.

Please **substitute** the paragraph starting on page 35, line 20,and ending on page 36, line 12 with the following paragraph:

In the bus-based system of this invention, a mechanism is provided to give each device on the bus a unique device identifier (device ID) after power-up or under other conditions as desired or needed by the system. A master can then use this device ID to access a specific device, particularly to set or modify registers 170 of the specified device, including the control and address registers. In the preferred embodiment, one master is assigned to carry out the entire system

configuration process. The master provides a series of unique device ID numbers for each unique device connected to the bus system. In the preferred embodiment, each device connected to the bus contains a special device-type register which specifies the type of device, for instance CPU, 4 MBit memory, 64 MBit memory or disk controller. The configuration master should check each device, determine the device type and set appropriate control registers, including access-time registers 173. The configuration master should check each memory device and set all appropriate memory address registers 172.

On page 36, **substitute** the paragraph starting on line 13, with the following paragraph:

One means to set up unique device ID numbers is to have each device to select a device ID in sequence and store the value in an internal device ID register 171. For example, a master can pass sequential device ID numbers through shift registers in each of a series of devices, or pass a token from device to device whereby the device with the token reads in device ID information from another line or lines. In a preferred embodiment, device ID numbers are assigned to devices according to their physical relationship, for instance, their order along the bus.

Please **substitute** the paragraph starting on page 38, line 25, and ending on page 39, line 12, with the following paragraph:

The configuration master should choose and set an access time in each access-time register 173 in each slave to a period sufficiently long to allow the slave to perform an actual, desired memory-access. For example, for a normal DRAM access, this time must be longer than the row address strobe (RAS) access time. If this condition is not met, the slave may not deliver the correct data. The value stored in a slave access-time register 173 is preferably one-half the number of bus cycles for which the slave device should wait before using the bus in response to a request. Thus an access time value of `1` would indicate that the slave should not access the bus until at least two cycles after the last byte of the request packet has been received. The value

of AccessReg0 is preferably fixed at 8 (cycles) to facilitate access to control registers.

Please **substitute** the paragraph starting on page 40, line 19, and ending on page 41, line 16, with the following paragraph:

In a preferred embodiment, a standard data block size can be selected for use with ECC, and the ECC method will determine the required number of bits of information in a corresponding ECC block. RAMs containing ECC information can be programmed to store an access time that in equal to: (1) the access time of the normal RAM (containing data) plus the time to access a standard data block (for corrected data) minus the time to send a request packet (6 bytes); or (2) the access time of a normal RAM minus the time to access a standard ECC block minus the time to send a request packet. To read a data block and the corresponding ECC block, the master simply issues a request for the data immediately followed by a request for the ECC block. The ECC RAM will wait for the selected access time then drive its data onto the bus right after (in case (1) above)) the data RAM has finished driving out the data block. Persons skilled in the art will recognize that the access time described in case (2) above can be used to drive ECC data before the data is driven onto the bus lines and will recognize that writing data can be done by analogy with the method described for a read. Persons skilled in the art will also recognize the adjustments that must be made in the bus-busy structure and the request packet arbitration methods of this invention in order to accommodate these paired ECC requests.

Please **substitute** the paragraph starting on page 45, line 17, and ending on page 46, line 17, with the following:

Referring to FIGS. 7a and 7b, although there is no stable condition where two devices drive the bus at the same time, conditions can arise because of propagation delay on the wires where one device, A 41, can start driving its part of the bus 44 while the bus is still being driven by another device, B 42 (already asserting a logical 1 on the bus). In a system using current drivers, when B 42 is driving the bus (before time 46), the value at points 44 and 45 is logical 1. If B

42 switches off at time 46 just when A 41 switches on, the additional drive by device A 41 causes the voltage at the output 44 of A 41 to briefly below the normal value. The voltage returns to its normal value at time 47 when the effect of device B 42 turning off is felt. The voltage at point 45 goes to logical 0 when device B 42 turns off, then drops at time 47 when the effect of device A 41 turning on is felt. Since the logical 1 driven by current from device A 41 is propagated irrespective of the previous value on the bus, the value on the bus is guaranteed to settle after one time of flight (t_f) delay, that is, the time it takes a signal to propagate from one end of the bus to the other. If a voltage drive was used (as in ECL wired-ORing), a logical 1 on the bus (from device B 42 being previously driven) would prevent the transition put out by device A 41 being felt at the most remote part of the system, e.g., device 43, until the turnoff waveform from device B 42 reached device A 41 plus one time of flight delay, giving a worst case settling time of twice the time of flight delay.

Please **substitute** the paragraph starting on page 46, line 20, and ending on page 47, line 12, with the following paragraph:

Clocking a high speed bus accurately without introducing error due to propagation delays can be implemented by having each device monitor two bus clock signals and then derive internally a device clock, the true system clock. The bus clock information can be sent on one or two lines to provide a mechanism for each bused device to generate an internal device clock with zero skew relative to all the other device clocks. Referring to FIG. 8a, in the preferred implementation, a bus clock generator 50 at one end of the bus propagates an early bus clock signal in one direction along the bus, for example on line 53 from right to left, to the far end of the bus. The same clock signal then is passed through the direct connection shown to a second line 54, and returns as a late bus clock signal along the bus from the far and to the origin, propagating from left to right. A single bus clock line can be used if it is left unterminated at the far end of the bus, allowing the early bus clock signal to reflect back along the same line as a late bus clock signal.

Please **substitute** the paragraph starting on page 49, line 12, and ending on page 50, line 3, with the following paragraph:

Referring to FIG. 9, each primary bus unit can be mounted on a single circuit board 66, sometimes called a memory stick. Each transceiver device 19 in turn connects to a transceiver bus 65, similar or identical in electrical and other respects to the primary bus 18 described at length above. In a preferred implementation, all masters are situated on the transceiver bus so there are no transceiver delays between masters and all memory devices are on primary bus units so that all memory accesses experience an equivalent transceiver delay, but persons skilled in the art will recognize how to implement systems which have masters on more than one bus unit and memory devices on the transceiver bus as well as on primary bus units. In general, each teaching of this invention which refers to a memory device can be practiced using a transceiver device and one or more memory devices on an attached primary bus unit. Other devices, generically referred to as peripheral devices, including disk controllers, video controllers or I/O devices can also be attached to either the transceiver bus or a primary bus unit, as desired. Persons skilled in the art will recognize how to use a single primary bus unit or multiple primary bus units as needed with a transceiver bus in certain system designs.

Please **substitute** the paragraph starting on page 53, line 24, and ending on page 54, line 22, with the following paragraph:

A block diagram of the preferred input/output circuit for address/data/control lines is shown in FIG. 10. This circuitry is particularly well-suited for use in DRAM devices but it can be used or modified by one skilled in the art for use in other devices connected to the bus of this invention. It consists of a set of input receivers 71, 72 and output driver 76 connected to input/output line 69 and pad 75 and circuitry to use the internal clock 73 and internal clock complement 74 to drive the input interface. The clocked input receivers take advantage of the synchronous nature of the bus. To further reduce the performance requirements for device input receivers, each device pin, and thus each bus line, is connected to two clocked receivers, one to sample the even cycle inputs, the other to sample the odd cycle

inputs. By thus de-multiplexing the input 69 at the pin, each clocked amplifier is given a full 2 ns cycle to amplify the bus low-voltage-swing signal into a full value CMOS logic signal. Persons skilled in the art will recognize that additional clocked input receivers can be used within the teachings of this invention. For example, four input receivers could be connected to each device pin and clocked by a modified internal device clock to transfer sequential bits from the bus to internal device circuits, allowing still higher external bus speeds or still longer settling times to amplify the bus low-voltage-swing signal into a full value CMOS logic signal.

Please **substitute** the paragraph starting on page 55, line 17, and ending on page 56, line 6, with the following paragraph:

The input receivers of every slave must be able to operate during every cycle to determine whether the signal on the bus is a valid request packet. This requirement leads to a number of constraints on the input circuitry. In addition to requiring small acquisition and resolution delays, the circuits must take little or no DC power, little AC power and inject very little current back into the input or reference lines. The standard clocked DRAM sense amp shown in FIG. 11 satisfies all these requirements except the need for low input currents. When this sense amp goes from sense to sample, the capacitance of the internal nodes 83 and 84 in FIG. 11 is discharged through the reference line 68 and input 69, respectively. This particular current is small, but the sum of such currents from all the inputs into the reference lines summed over all devices can be reasonably large.

Please **substitute** the paragraph starting on page 58, line 13, and ending on 59, line 2, with the following paragraph:

In the preferred embodiment, two sets of these delay lines are used, one to generate the true value of the internal device clock 73, and the other to generate the complement 74 without adding any inverter delay. The dual circuit allows generation of truly complementary clocks, with extremely small skew. The complement internal device clock is used to clock the `event` input receivers to sample at time 127,

while the true internal device clock is used to clock the `odd` input receivers to sample at time 125. The true and complement internal device clocks 73 and 74 are also used to select which data is driven to the output drivers. The gate delay between the internal device clock and output circuits driving the bus is slightly greater than the corresponding delay for the input circuits, which means that the new data always will be driven on the bus slightly after the old data has been sampled.

Please **<u>substitute</u>** the paragraph starting on page 60, line 1, and ending on page 61, line 8, with the following paragraph:

Running the internal I/O lines in the conventional way at high bus cycle rates is not possible. In the preferred method, several (preferably 4) bytes are read or written during each cycle and the column access path is modified to ran at a lower rate (the inverse of the number of bytes accessed per cycle, preferably 1/4 of the bus cycle rate). Three different techniques are used to provide the additional internal I/O lines required and to supply data to memory cells at this rate. First, the number of I/O bit lines in each subarray running through the column decoder 147 A, B is increased, for example, to 16, eight for each of the two columns of column sense amps and the column decoder selects one set of columns from the "top" half 148 of subarray 150 and one set of columns from the "bottom" half 149 during each cycle, where the column decoder selects one column sense amp per I/O bit line. Second, each column I/O line is divided into two halves, carrying data independently over separate internal I/O lines from the left half 147A and right half 147B of each subarray (dividing each subarray into quadrants) and the column decoder selects sense amps from each right and left half of the subarray, doubling the number of bits available at each cycle. Thus each column decode selection turns on n column sense amps, where n equals four (top left and right, bottom left and right quadrants) times the number of I/O lines in the bus to each subarray quadrant (8 lines each times 4=32 lines in the preferred implementation). Finally, during each RAS cycle, two different subarrays, e.g. 157 and 153, are accessed. This doubles again the available number of I/O lines containing data. Taken together, these

changes increase the internal I/O bandwidth by at least a factor of 8. Four internal buses are used to route these internal I/O lines. Increasing the number of I/O lines and then splitting them in the middle greatly reduces the capacitance of each internal I/O line which in turn reduces the column access time, increasing the column access bandwidth even further.

IN THE CLAIMS:

Kindly **cancel** claims 1-150, without prejudice.

Kindly **add** the following claims:

--151. A synchronous memory device including an array of memory cells, the synchronous memory device comprises:

clock receiver circuitry to receive an external clock signal;

input receiver circuitry to sample a first operation code synchronously with respect to a transition of the external clock signal;

a programmable register to store a value which is representative of an amount of time to transpire before the memory device outputs data, wherein the memory device stores the value in the programmable register in response to the first operation code; and

output driver circuitry to output data in response to a second operation code, wherein the data is output after the amount of time transpires, and wherein:

> the output driver circuitry outputs a first portion of the data synchronously with respect to a rising edge transition of the external clock signal and outputs a second portion of the data synchronously with respect to a falling edge transition of the external clock signal.

152. The memory device of claim 151 wherein the first operation code is included in a control register access packet.

153. The memory device of claim 152 wherein the first operation code and the value are included in the same control register access packet.

154. The memory device of claim 151 wherein the memory device is a synchronous DRAM.

155. The memory device of claim 151 wherein the input receiver circuitry receives the second operation code and address information corresponding to at least one memory cell location of the array of memory cells.

156. The memory device of claim 151 wherein the input receiver circuitry receives at least one of the second operation code and

address information corresponding to at least one memory cell location of the array of memory cells.

157. The memory device of claim 151 wherein the amount of time is representative of a number of clock cycles of the external clock signal.

158. The memory device of claim 151 wherein the input receiver circuitry receives a third operation code, wherein the third operation code initiates a write operation in the memory device.

159. The memory device of claim 158 wherein the input receiver circuitry receives the third operation code and address information corresponding to at least one memory cell location of the array of memory cells.

160. The memory device of claim 151 further including delay lock loop circuitry coupled to the clock receiver circuitry to generate a first internal clock signal, wherein the data is output using the first internal clock signal.

161. The memory device of claim 151 wherein the output driver circuitry outputs the data onto a bus.

162. The memory device of claim 161 wherein the bus includes a set of signal lines used to transmit multiplexed address information, data and control information.

163. A method of operation of a synchronous memory device, wherein the memory device includes an array of memory cells and a programmable register, the method of operation of the memory device comprises:

sampling a first operation code synchronously with respect to an external clock signal;

receiving a binary value wherein the memory device stores the binary value in the programmable register in response to the first operation code wherein the binary value is representative of an amount

of time to transpire before the memory device outputs data in response to a second operation code;

sampling the second operation code; and

outputting data after the amount of time transpires, wherein a first portion of the data is output synchronously with respect to a first transition of the external clock signal and a second portion of the data is output synchronously with respect to a second transition of the external clock signal.

164. The method of claim 163 wherein the second operation code is sampled synchronously with respect to the external clock signal.

165. The method of claim 163 wherein the binary value is representative of a number of clock cycles of the external clock signal.

166. The method of claim 165 further including:

receiving block size information wherein the block size information defines an amount of data to be output in response to the second operation code, wherein the memory device outputs the amount of data after the number of clock cycles of the external clock signal transpire.

167. The method of claim 163 further including receiving address information synchronously with respect to the external clock signal.

168. The method of claim 163 wherein the address information and the second operation code are included in a read request packet.

169. The method of claim 163 further including receiving precharge information.

170. The method of claim 169 wherein the precharge information includes a binary bit, wherein, after accessing the data from the array of memory cells, the memory device retains contents of a plurality of

sense amplifiers for a subsequent memory operation as a result of a first state of the binary bit.

171. The method of claim 163 wherein the first transition of the external clock signal is a rising edge transition and the second transition of the external clock signal is a falling edge transition.

172. The method of claim 171 wherein the first and second transitions of the external clock signal are consecutive transitions of the external clock signal.

173. The method of claim 163 wherein the first operation code is sampled during an initialization sequence after power is applied to the memory device.

174. The method of claim 163 wherein the memory device samples the first operation code from an external bus, and wherein the memory device outputs data onto the external bus.

175. The method of claim 174 wherein the external bus includes a set of signal lines used to transmit multiplexed address information, data and control information.

176. A method of controlling a synchronous memory device by a memory controller, wherein the memory device includes an array of memory cells and a programmable register, the method of controlling the memory device comprises:

issuing a first operation code to the memory device, wherein the first operation code initiates an access of the programmable register in the memory device in order to store a binary value;

providing the binary value to the memory device, wherein the memory device stores the binary value in the programmable register in response to the first operation code;

issuing a second operation code to the memory device, wherein the second operation code instructs the memory device to accept data that is issued by the memory controller;

issuing a first portion of the data to the memory device synchronously with respect to a first transition of the external clock signal; and

issuing a second portion of the data to the memory device synchronously with respect to a second transition of the external clock signal.

177. The method of claim 176 wherein the binary value is representative of a delay time to transpire before the memory device samples the data, and wherein the first portion of the data is issued to the memory device after the delay time transpires.

178. The method of claim 176 wherein the binary value is representative of a number of clock cycles of the external clock signal.

179. The method of claim 176 wherein the first transition of the external clock signal is a rising edge transition and the second transition of the external clock signal is a falling edge transition.

180. The method of claim 176 further including:

providing block size information to the memory device, wherein the block size information defines an amount of data to be accepted by the memory device in response to the second operation code.

181. The method of claim 176 further including providing address information to the memory device.

182. The method of claim 181 wherein the address information and the second operation code are included in a write request packet.

183. The method of claim 176 wherein the first operation code and the data are issued onto an external bus.

184. The method of claim 183 wherein the external bus includes a set of signal lines used to transmit multiplexed address information, data and control information.

185. The method of claim 176 wherein the second operation code includes precharge information.

186. A synchronous memory device, wherein the memory device includes an array of memory cells, the memory device comprises:

input receiver circuitry to sample a first operation code synchronously with respect to an external clock signal;

a programmable register to store a binary value in response to the first operation code, wherein the binary value is representative of an amount of time to transpire before the memory device outputs data; and

output driver circuitry to output data in response to a second operation code and after the amount of time transpires, wherein a first portion of the data is output synchronously with respect to a first transition of the external clock signal and a second portion of the data is output synchronously with respect to a second transition of the external clock signal.

187. The memory device of claim 186 wherein the binary value is representative of a number of clock cycles of the external clock signal.

188. The memory device of claim 187 wherein the binary value is representative of a fractional number of clock cycles of the external clock signal.

188. The memory device of claim 186 wherein the first transition of the external clock signal is a rising edge transition and the second transition of the external clock signal is a falling edge transition.

189. The memory device of claim 188 wherein the first and second transitions of the external clock signal are consecutive transitions.

190. The memory device of claim 189 wherein the first operation code and the binary value are included in a packet.

191. The memory device of claim 190 wherein the first operation code and the binary value are included in the same packet.

192. The memory device of claim 186 further including delay lock loop circuitry to generate a first internal clock signal, wherein the data is output using the first internal clock signal.

193. The memory device of claim 186 wherein the input receiver circuitry receives address information corresponding to at least one memory cell location of the array of memory cells.

194. The memory device of claim 186 wherein the output driver circuitry outputs data onto an external bus having a set of signal lines used to transmit multiplexed address information, data and control information.

195. The memory device of claim 194 wherein the input receiver circuitry samples the first operation code from the external bus.

196. The memory device of claim 186 wherein the output driver circuitry and the input receiver circuitry are connected to a common pad.

197. The memory device of claim 186 wherein the memory device is a synchronous DRAM.--

REMARKS

This Preliminary Amendment seeks to place this application in condition for allowance. This application is a continuation of Application No. 09/545,648 which is a continuation of Application No. which is a continuation of Application No. 09/161,090. Application Serial No. 09/545,648 is pending.

REQUEST FOR PRIORITY

Applicants request priority to Application Serial No. 07/510,898, filed April 18, 1990, now abandoned. Applicants request such priority through Application No. 09/545,648, filed on April 10, 2000 (still pending), which is a continuation of Application No. 09/161,090, filed on September 25, 1998 (now U.S. Patent 6,049,846), which is a division of Application No. 08/798,520, filed on February 10, 1997 (now U.S. Patent 5,841,580), which is a division of Application No. 08/448,657, filed May 24, 1995 (now U.S. Patent 5,638,334), which is a division of Application No. 08/222,646, filed on March 31, 1994 (now U.S. Patent 5,513,327), which is a continuation of Application No. 07/954,945, filed on September 30, 1992 (now U.S. Patent 5,319,755), which is a continuation of Application No. 07/510,898, filed on April 18, 1990 (now abandoned).

Accordingly, Applicants claim the benefit of the filing date of Application Serial No. 07/510,898 -- i.e., April 18, 1990. The specification has been amended to identify the continuation or related U.S. application data identified above. No new matter has been added.

AMENDMENTS TO THE CLAIMS

In this continuation application, Applicants present new claims which set forth novel and unobvious features of Applicants' invention. Applicants submit new claims 151-197 to more fully protect the Applicants invention. No new matter has been added.

The newly submitted claims are believed to be fully supported by the specification -- see, for example, Figures 1, 2, 4, and 10-13; page 14, line 3 to page 16, line 7; page 18, line 22 to page 21, line 20;

page 22, line 11 to page 25, line 8; page 27, line 1 to page 28, line 20; page 48, lines 6-17; page 53, line 5 to page 59, line 2; page 71, line 14 to page 72, line 21;and page 115, lines 10-22.

AMENDMENTS TO THE SPECIFICATION

Applicants have also amended the specification to include the priority data, reflect changes to the drawings and to correct obvious spelling, typographical and grammatical errors. No new matter has been added.

AMENDMENTS TO THE ABSTRACT

A new Abstract of the Disclosure is attached hereto. No new matter has been added.

DRAWING CHANGES

Accompanying this Preliminary Amendment is a Request to Approve Drawing Changes. Applicants have amended the drawings to show every feature of the invention specified in the claims. To that end, Applicants submit herewith new Figure 16. A copy of Applicants Request to Approve Drawing Changes is attached.

New Figure 16 is added to illustrate, among other things, access-time register(s) 173. Figure 16 illustrates one embodiment of the internal registers within each device illustrated in Figure 2. Support may be found in the specification at page 14, lines 3-21 and page 53 lines 4-21. No new matter has been added.

Applicants seek to amend Figure 10 to more fully reflect the discussion in the specification, in particular, page 55, lines 12-16 and page 58, lines 13-23. The proposed changes are indicated in red. No new matter has been added. Applicants respectfully request that the Examiner approve the proposed changes to Figure 10. A new Figure 10 which incorporates the changes is also attached to the Request.

CONCLUSION

Applicants request entry of the foregoing amendment prior to examination of this application. Applicants submit that all of the claims present patentable subject matter. Accordingly, Applicants respectfully request allowance of all of the claims.

Respectfully submitted,

Date: April 12, 2001

Neil A. Steinberg
Reg. No. 34,735
650-947-5325

ABSTRACT OF THE DISCLOSURE

A synchronous memory device and methods of operation and controlling such a device. The synchronous memory device includes input receiver circuitry to sample a first operation code synchronously with respect to an external clock signal. The synchronous memory device also includes a programmable register to store a binary value in response to the first operation code, wherein the binary value is representative of an amount of time to transpire before the memory device outputs data. The synchronous memory device also includes output driver circuitry to output data in response to a second operation code and after the amount of time transpires. A first portion of data is output synchronously with respect to a first transition of the external clock signal and a second portion of data is output synchronously with respect to a second transition of the external clock signal.

Version With Markings to show Changes made to the Specification

On page 3, starting on line 6:

Prior art memory systems have attempted to solve the problem of high speed access to memory with limited success. U.S. Pat. No. 3,821,715 (Hoff et.al.), was issued to Intel Corporation for the earliest 4-bit [micro-processor] microprocessor. That patent describes a bus connecting a single central processing unit (CPU) with multiple RAMs and ROMs. That bus multiplexes addresses and data over a 4-bit wide bus and uses point-to-point control signals to select particular RAMs or ROMs. The access time is fixed and only a single processing element in permitted. There is no block-mode type of operation, and most important, not all of the interface signals between the devices are bused (the ROM and RAM control lines and the RAM select lines are point-to-point).

On page 6, starting on line 1:

In U.S. Pat. No. [4,646,270] 4,646,279(Voss), a video RAM is described which implements a parallel-load, serial-out shift register on the output of a DRAM. This generally allows greatly improved bandwidth (and has been extended to 2, 4 and greater width shift-out paths.) The rest of the interfaces to the DRAM (RAS, CAS, multiplexed address, etc.) remain the same as for conventional DRAMS.

On page 10, starting on lines 18:

[Figure 7 shows] FIGS. 7a and 7b show the timing whereby signals from two devices can overlap temporarily and drive the bus at the same time.

[Figure 8 shows] FIGS. 8a and 8b show the connection and timing between bus clocks and devices on the bus.

On page 14, starting on line 3:

[Each] With reference to FIG. 16, each semiconductor device contains a set of internal registers 170, preferably including a device identification (device register 171, a device-type descriptor register 174, control registers 175 and other registers containing other

information relevant to that type of device. In a preferred implementation, semiconductor devices connected to the bus contain registers 172 which specify the memory addresses contained within that device and access-time registers 173 which store a set of one or more delay times at which the device can or should be available to send or receive data.

On page 14, starting on line 13:

Most of these registers can be modified and preferably are set as part of an initialization sequence that occurs when the system is powered up or reset. During the initialization sequence each device on the bus is assigned a unique device ID number, which is stored is the device ID register 171. A bus master can then use these device ID numbers to access and set appropriate registers in other devices, including access-time to registers 173, control registers 175, and memory registers 172, to configure the system. Each slave may have one or several access-time registers 173 (four in a preferred embodiment). In a preferred embodiment, one access-time register in each slave is permanently or semi-permanently programmed with a fixed value to facilitate certain control functions. A preferred implementation of an initialization sequence is described below in more detail.

On page 21, starting on line 8:

The data block transfer occurs later at a time specified in the request packet control information, preferably beginning on an even cycle. A device begins a data block transfer almost immediately with a device-internal phase as the device initiates certain functions, such as setting up memory addressing, before the bus access phase begins. The time after which a data block is driven onto the bus lines is selected from values stored in slave access-time registers 173. The timing of data for reads and writes in preferably the same; the only difference is which device drives the bus. For reads, the slave drives the bus and the master latches the values from the bus. For writes the master drives the bus and the selected slave latches the values from the bus.

On page 34, starting on line 4:

Slave devices <u>do</u> not need to detect a collision directly, but they must wait to do anything irrecoverable until the last byte (byte 5) is read to ensure that the packet is valid. A request packet with Master[0:3] equal to 0 (a retry signal) is ignored and does not cause a collision. The subsequent bytes of such a packet are ignored.

On page 35, starting on line 20:

In the bus-based system of this invention, a mechanism is provided to give each device on the bus a unique device identifier (device ID) after power-up or under other conditions as desired or needed by the system. A master can then use this device ID to access a specific device, particularly to set or modify registers <u>170</u> of the specified device, including the control and address registers. In the preferred embodiment, one master is assigned to carry out the entire system configuration process. The master provides a series of unique device ID numbers for each unique device connected to the bus system. In the preferred embodiment, each device connected to the bus contains a special device-type register which specifies the type of device, for instance CPU, 4 MBit memory, 64 MBit memory or disk controller. The configuration master should check each device, determine the device type and set appropriate control registers, including access-time registers 173. The configuration master should check each memory device and set all appropriate memory address registers 172.

On page 36, starting on line 13:

One means to set up unique device ID numbers is to have each device to select a device ID in sequence and store the value in an internal device ID register <u>171</u>. For example, a master can pass sequential device ID numbers through shift registers in each of a series of devices, or pass a token from device to device whereby the device with the token reads in device ID information from another line or lines. In a preferred embodiment, device ID numbers are assigned to devices according to their physical relationship, for instance, their order along the bus.

On page 38, starting on line 25:

The configuration master should choose and set an access time in each access-time register 173 in each slave to a period sufficiently long to allow the slave to perform an actual, desired memory-access. For example, for a normal DRAM access, this time must be longer than the row address strobe (RAS) access time. If this condition is not met, the slave may not deliver the correct data. The value stored in a slave access-time register 173 is preferably one-half the number of bus cycles for which the slave device should wait before using the bus in response to a request. Thus an access time value of `1` would indicate that the slave should not access the bus until at least two cycles after the last byte of the request packet has been received. The value of AccessReg0 is preferably fixed at 8 (cycles) to facilitate access to control registers.

On page 40, starting on line 19:

In a preferred embodiment, a standard data block size can be selected for use with ECC, and the ECC method will determine the required number of bits of information in a corresponding ECC block. RAMs containing ECC information can be programmed to store an access time that in equal to: (1) the access time of the normal RAM (containing data) plus the time to access a standard data block (for corrected data) minus the time to send a request packet (6 bytes); [or']or (2) the access time of a normal RAM minus the time to access a standard ECC block minus the time to send a request packet. To read a data block and the corresponding ECC block, the master simply issues a request for the data immediately followed by a request for the ECC block. The ECC RAM will wait for the selected access time then drive its data onto the bus right after (in case (1) above)) the data RAM has finished driving out the data block. Persons skilled in the art will recognize that the access time described in case (2) above can be used to drive ECC data before the data is driven onto the bus lines and will recognize that writing data can be done by analogy with the method described for a read. Persons skilled in the art will also recognize the adjustments that must be made in the bus-busy structure and the

request packet arbitration methods of this invention in order to accommodate these paired ECC requests.

On page 45, starting on line 17:

Referring to [Fig. 7] FIGS. 7a and 7b, although there is no stable condition where two devices drive the bus at the same time, conditions can arise because of propagation delay on the wires where one device, A 41, can start driving its part of the bus 44 while the bus is still being driven by another device, B 42 (already asserting a logical 1 on the bus). In a system using current drivers, when B 42 is driving the bus (before time 46), the value at points 44 and 45 is logical 1. If B 42 switches off at time 46 just when A 41 switches on, the additional drive by device A 41 causes the voltage at the output 44 of A 41 to briefly below the normal value. The voltage returns to its normal value at time 47 when the effect of device B 42 turning off is felt. The voltage at point 45 goes to logical 0 when device B 42 turns off, then drops at time 47 when the effect of device A 41 turning on is felt. Since the logical 1 driven by current from device A 41 is propagated irrespective of the previous value on the bus, the value on the bus is guaranteed to settle after one time of flight (t_f) delay, that is, the time it takes a signal to propagate from one end of the bus to the other. If a voltage drive was used (as in ECL wired-ORing), a logical 1 on the bus (from device B 42 being previously driven) would prevent the transition put out by device A 41 being felt at the most remote part of the system, e.g., device 43, until the turnoff waveform from device B 42 reached device A 41 plus one time of flight delay, giving a worst case settling time of twice the time of flight delay.

On page 46, starting on line 20:

Clocking a high speed bus accurately without introducing error due to propagation delays can be implemented by having each device monitor two bus clock signals and then derive internally a device clock, the true system clock. The bus clock information can be sent on one or two lines to provide a mechanism for each bused device to generate an internal device clock with zero skew relative to all the other device clocks. Referring to FIG. 8a, in the preferred implementation, a bus

clock generator 50 at one end of the bus propagates an early bus clock signal in one direction along the bus, for example on line 53 [from left to right] from right to left, to the far end of the bus. The same clock signal then is passed through the direct connection shown to a second line 54, and returns as a late bus clock signal along the bus from the far and to the origin, propagating from [right] left to [left] right. A single bus clock line can be used if it is left unterminated at the far end of the bus, allowing the early bus clock signal to reflect back along the same line as a late bus clock signal.

On page 49, starting on line 12:

Referring to FIG. 9, each primary bus unit can be mounted on a single circuit board 66, sometimes called a memory stick. Each transceiver device 19 in turn connects to a transceiver bus 65, similar or identical in electrical and other respects to the primary bus 18 described at length above. In a preferred implementation, all masters are situated on the transceiver bus so there are no transceiver delays between masters and all memory devices are on primary bus units so that all memory accesses experience an equivalent transceiver delay, but persons skilled in the art will recognize how to implement systems which have masters on more than one bus unit and memory devices on the transceiver bus as well as on [primay] primary bus units. In general, each teaching of this invention which refers to a memory device can be practiced using a transceiver device and one or more memory devices on an attached primary bus unit. Other devices, generically referred to as peripheral devices, including disk controllers, video controllers or I/O devices can also be attached to either the transceiver bus or a primary bus unit, as desired. Persons skilled in the art will recognize how to use a single primary bus unit or multiple primary bus units as needed with a transceiver bus in certain system designs.

On page 53, starting on line 24:

A block diagram of the preferred input/output circuit for address/data/control lines is shown in FIG. 10. This circuitry is particularly well-suited for use in DRAM devices but it can be used or modified by one skilled in the art for use in other devices connected

to the bus of this invention. It consists of a set of input receivers 71, 72 and output driver 76 connected to input/output line 69 and pad 75 and circuitry to use the internal clock 73 and internal clock complement 74 to drive the input interface. The clocked input receivers take advantage of the synchronous nature of the bus. To further reduce the performance requirements for device input receivers, each device pin, and thus each bus line, is connected to two clocked receivers, one to sample the even cycle inputs, the other to sample the odd cycle inputs. By thus de-multiplexing the input [70]69 at the pin, each clocked amplifier is given a full 2 ns cycle to amplify the bus low-voltage-swing signal into a full value CMOS logic signal. Persons skilled in the art will recognize that additional clocked input receivers can be used within the teachings of this invention. For example, four input receivers could be connected to each device pin and clocked by a modified internal device clock to transfer sequential bits from the bus to internal device circuits, allowing still higher external bus speeds or still longer settling times to amplify the bus low-voltage-swing signal into a full value CMOS logic signal.

On page 55, starting on line 17:

The input receivers of every slave must be able to operate during every cycle to determine whether the signal on the bus is a valid request packet. This requirement leads to a number of constraints on the input circuitry. In addition to requiring small acquisition and resolution delays, the circuits must take little or no DC power, little AC power and inject very little current back into the input or reference lines. The standard clocked DRAM sense amp shown in FIG. 11 satisfies all these requirements except the need for low input currents. When this sense amp goes from sense to sample, the capacitance of the internal nodes 83 and 84 in [figure11] FIG. 11 is discharged through the reference line 68 and input 69, respectively. This particular current is small, but the sum of such currents from all the inputs into the reference lines summed over all devices can be reasonably large.

On page 58, starting on line 13:

In the preferred embodiment, two sets of these delay lines are used, one to generate the true value of the internal device clock 73, and the other to generate the complement 74 without adding any inverter delay. The dual circuit allows generation of truly complementary clocks, with extremely small skew. The complement internal device clock is used to clock the `event` input receivers to sample at time 127, while the true internal device clock is used to clock the `odd` input receivers to sample at time 125. The true and complement internal device clocks 73 and 74 are also used to select which data is driven to the output drivers. The gate delay between the internal device clock and output circuits driving the bus is slightly greater than the corresponding delay for the input circuits, which means that the new data always will be driven on the bus slightly after the old data has been sampled.

Starting on page 60, line 1:

Running the internal I/O lines in the conventional way at high bus cycle rates is not possible. In the preferred method, several (preferably 4) bytes are read or written during each cycle and the column access path is modified to ran at a lower rate (the inverse of the number of bytes accessed per cycle, preferably 1/4 of the bus cycle rate). Three different techniques are used to provide the additional internal I/O lines required and to supply data to memory cells at this rate. First, the number of I/O bit lines in each subarray running through the column decoder 147 A, B is increased, for example, to 16, eight for each of the two columns of column sense amps and the column decoder selects one set of columns from the "top" half 148 of subarray 150 and one set of columns from the "bottom" half 149 during each cycle, where the column decoder selects one column sense amp per I/O bit line. Second, each column I/O line is divided into two halves, carrying data independently over separate internal I/O lines from the left half 147A and right half 147B of each subarray (dividing each subarray into quadrants) and the column decoder selects sense amps from each right and left half of the subarray, doubling the number of bits available at each cycle. Thus each column decode selection turns on n

column sense amps, where n equals four (top left and right, bottom left and right quadrants) times the number of I/O lines in the bus to each subarray quadrant (8 lines each times 4=32 lines in the preferred implementation). Finally, during each RAS cycle, two different subarrays, e.g. 157 and 153, are accessed. This doubles again the available number of I/O lines containing data. Taken together, these changes increase the internal I/O bandwidth by at least a factor of 8. Four internal buses are used to route these internal I/O lines. Increasing the number of I/O lines and then splitting them in the middle greatly reduces the capacitance of each internal I/O line which in turn reduces the column access time, increasing the column access bandwidth even further.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
(Case No. **RA043D2C15**)

In the Application of:		)
	FARMWALD ET AL.	)
Serial No:	Continuation of 09/545,648	)
Filed:	Herewith	)
Title:	MEMORY DEVICE HAVING A PROGRAMMABLE REGISTER (As Amended)	)

Assistant Commissioner for Patents
Washington, DC 20231

REQUEST TO APPROVE DRAWING CHANGES

Dear Sir:

Attached hereto is new Figure 16. Figure 16 illustrates the internal registers which reside in each device illustrated in Figure 2. This embodiment is described in the specification at page 14, lines 3-21 and page 53 lines 4-21. No new matter has been added.

Applicants seek to amend Figure 10 to more fully reflect the discussion in the specification, specifically, page 55, line 12-16 and page 58, lines 13-23. Also attached, is a photocopy of Figure 10 with the proposed changes indicated in red. No new matter has been added.

Applicants respectfully request that the proposed new Figure 16 be approved by the Examiner. Applicants also respectfully request approval of the proposed changes to Figure 10. A new Figure 10 which incorporates the changes is also attached hereto.

Respectfully submitted,

Date: April 12, 2001

Neil A. Steinberg
Reg. No. 34,735
650-947-5325

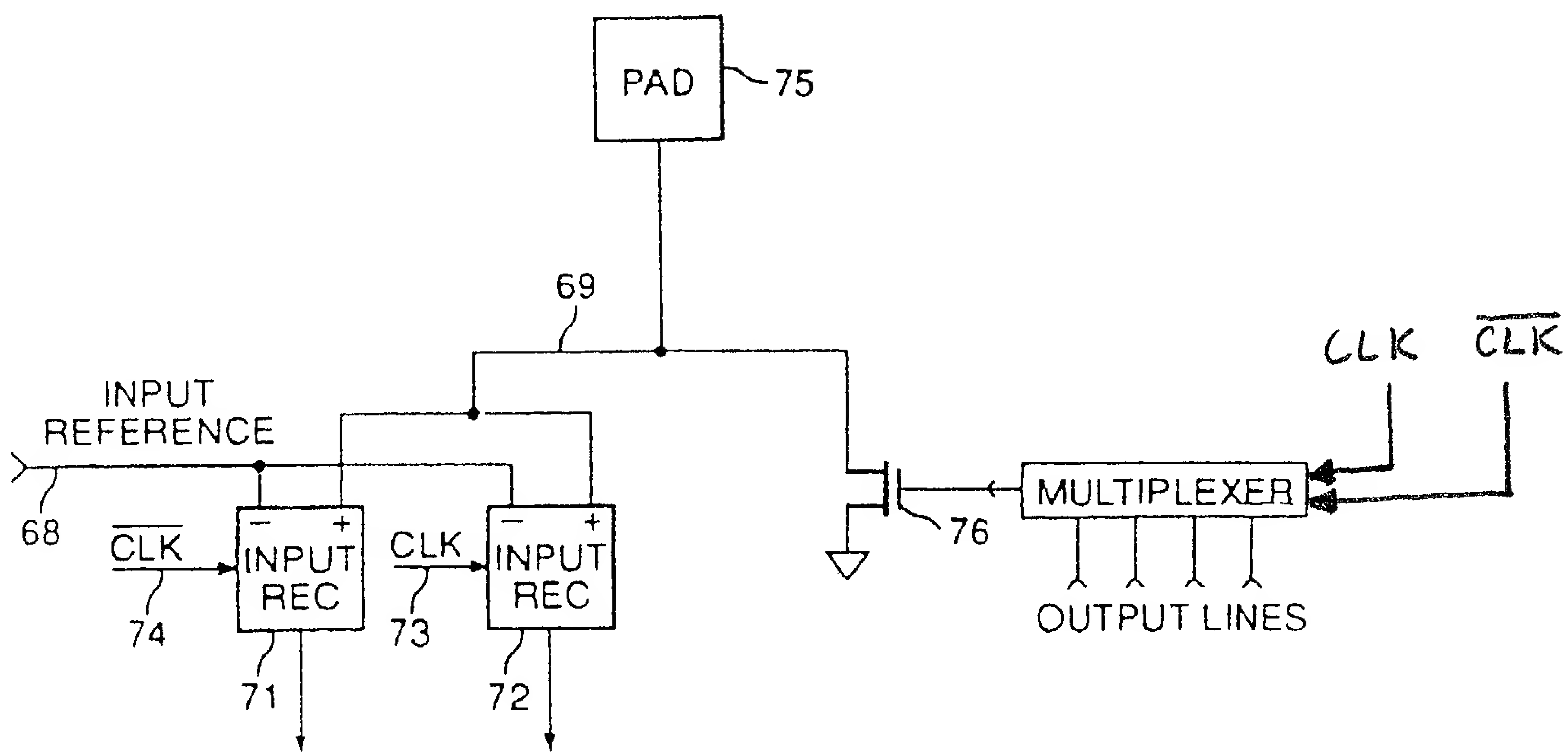
PAD
75
69
INPUT
REFERENCE
68
$\overline{CLK}$
74
INPUT
REC
71
CLK
73
INPUT
REC
72
76
MULTIPLEXER
OUTPUT LINES
CLK
$\overline{CLK}$

FIG. 10

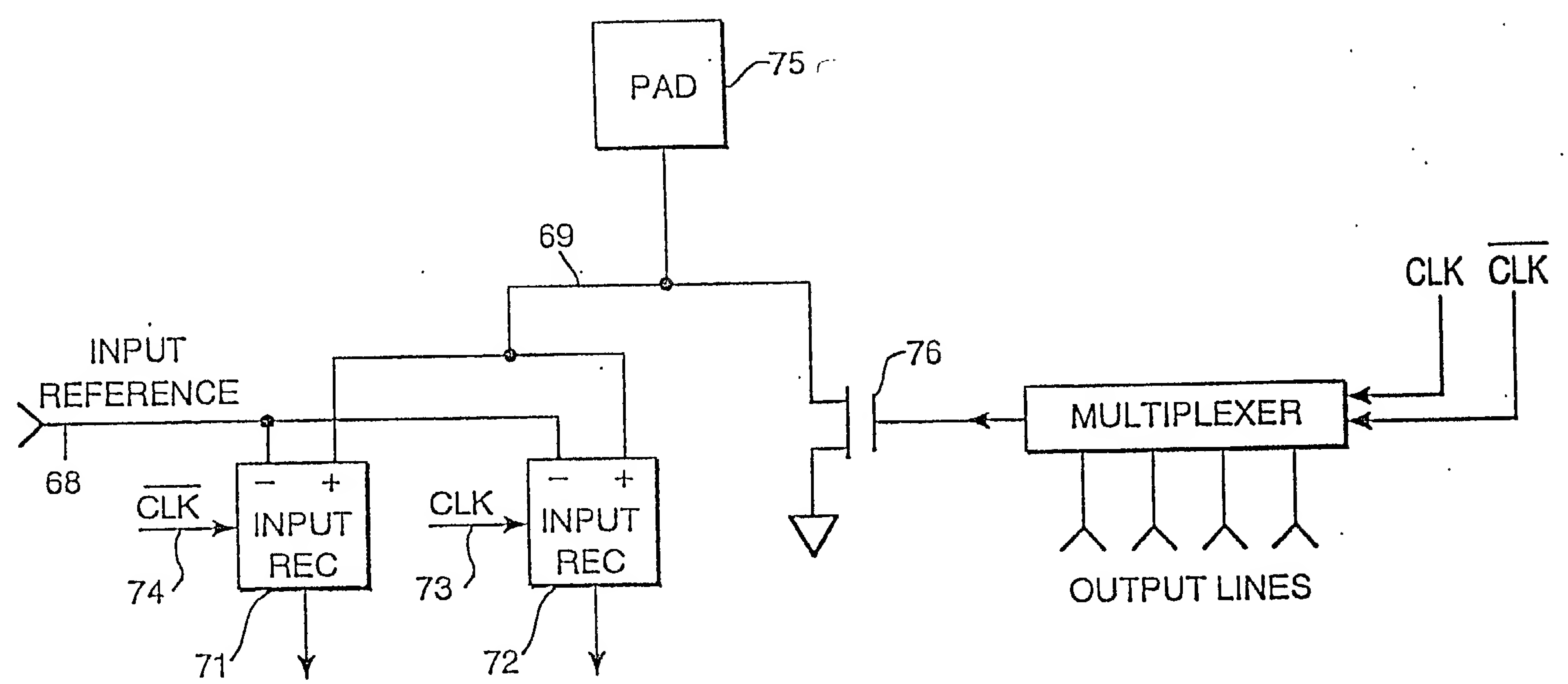
PAD
75
69
76
CLK
CLK
INPUT
REFERENCE
68
MULTIPLEXER
CLK
INPUT
REC
74
71
CLK
INPUT
REC
73
72
OUTPUT LINES

FIG_10

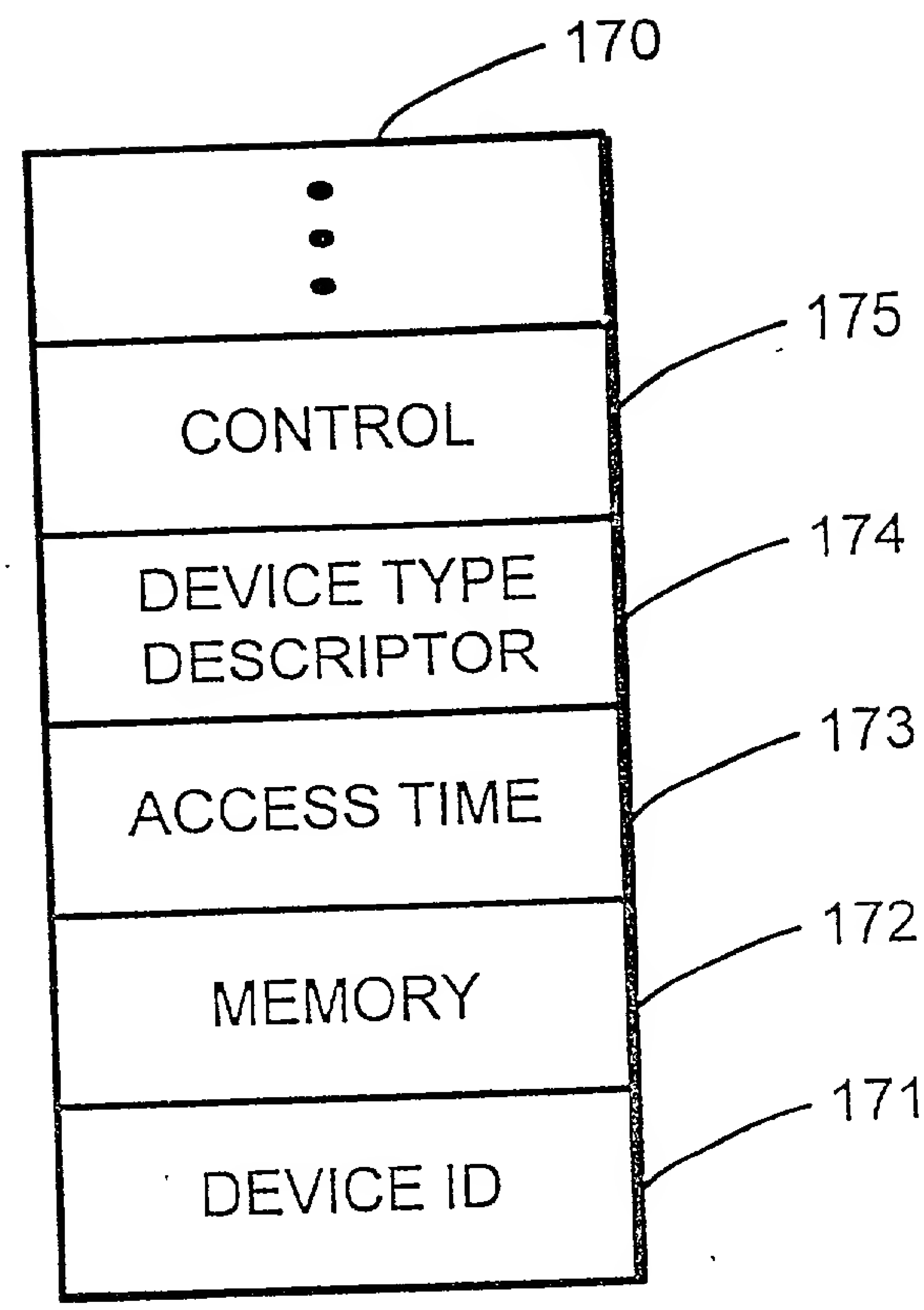
170
175
CONTROL
174
DEVICE TYPE DESCRIPTOR
173
ACCESS TIME
172
MEMORY
171
DEVICE ID

FIG. 16